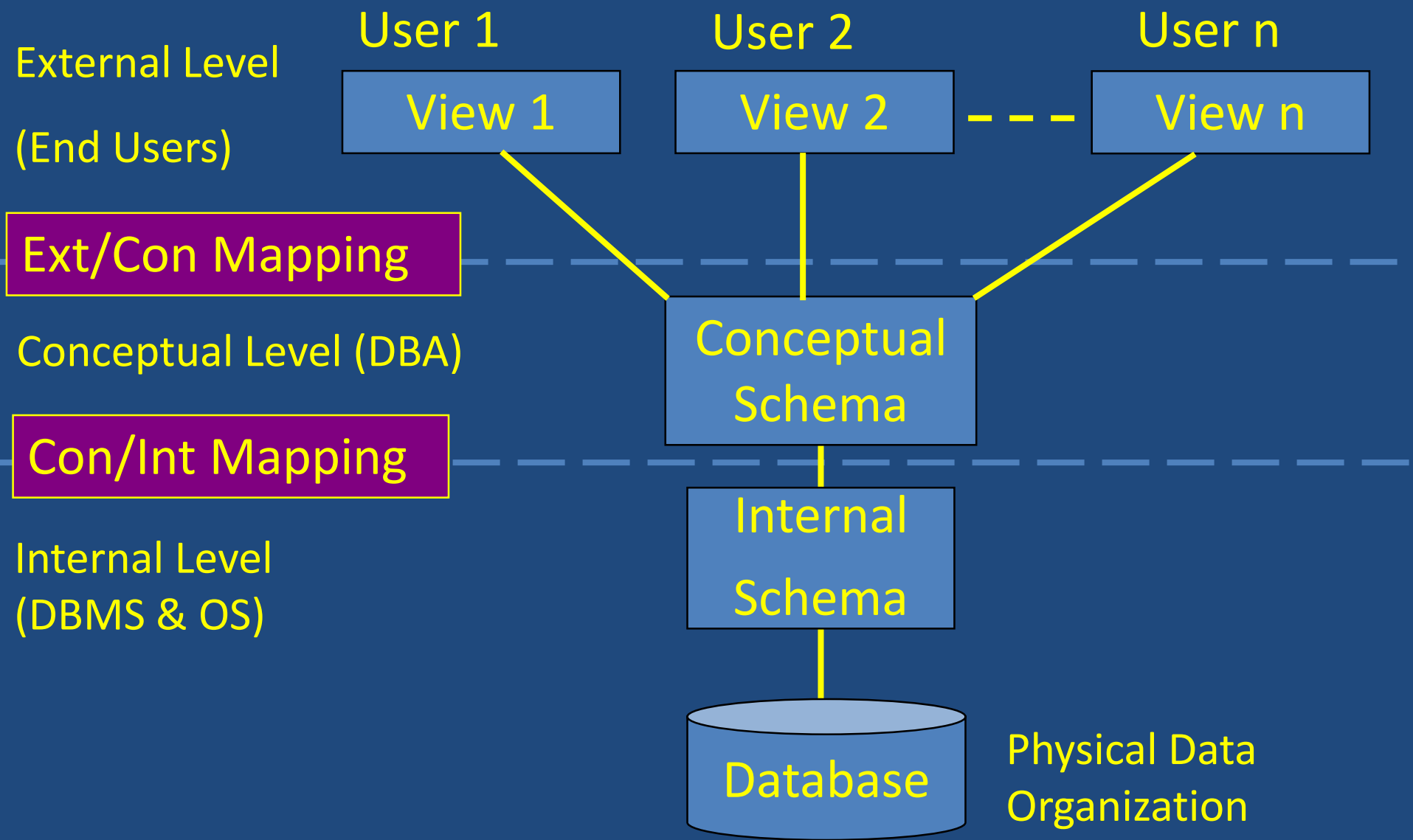


Three-Level Architecture for Database Development

- DBTG (Data Base Task Group), appointed by the Conference on Data Systems and Languages (CODASYL, 1971) proposed two-level approach:
 - A system view called the schema
 - User views called sub-schemas
- American National Standards Institute (ANSI) and Standards Planning and Requirements Committee (SPARC), ANSI/X3/SPARC, 1975 proposed three-level approach:
 - External Level
 - Conceptual Level
 - Internal Level

The Three-Level ANSI-SPARC Architecture



Differences between the three levels

First Name: Rana
 Last Name: Aslam
 Date of Birth:
 12 Sep, 1970



Name: R. Aslam
 Age: 24y,10d
 Dept: Sales

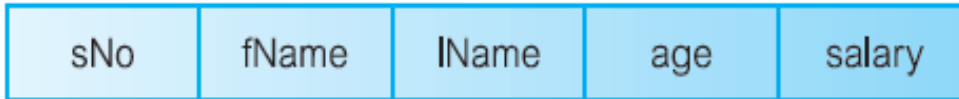
<u>Name</u>	<u>DoB</u>	<u>Deps</u>	<u>DepId</u>
Rana Aslam	12/09/70	5	D001
Marya Wasti	29/02/80	0	D005

BH | RH | Rana Aslam 120970 5 D001 | RH | Marya Wasti...

01110011010011100101001010100101010010101.....

Differences between the three levels

External view 1



External view 2



Conceptual level



Internal level

```
struct STAFF {
    int staffNo;
    int branchNo;
    char fName [15];
    char lName [15];
    struct date dateOfBirth;
    float salary;
    struct STAFF *next;           /* pointer to next Staff record */
};
index staffNo; index branchNo; /* define indexes for staff */
```

Objectives of Three-Level Architecture

- A standard for database systems
- A user's view is immune to changes made in other views
- Users should not need to know physical database storage details
- DBA should be able to change conceptual structure of database without affecting all users
- Internal structure of database should be unaffected by changes to physical aspects of storage

External Level/View

- The users' view of the database
- This level describes the part of the database that is relevant to each user

Conceptual Level/View

- The community view of the database
- Describes what data is stored in the database and the relationships among the data
- Contains the logical structure of the entire database as seen by the DBA

Conceptual Level/View

- Must not contain any storage-dependent details
- For instance, the description of an entity should contain only data types of attributes (for example, integer, real, character) and their length (such as the maximum number of digits or characters), but not any storage considerations, such as the number of bytes occupied

Internal Level/View

- The way the DBMS and the operating system perceive the data
- The physical representation of the database on the computer
- Describes how the data is stored in the database
- There is only one conceptual schema and one internal schema per database

Data Independence

- A major outcome of 3-L Arch
- Changes in lower level do not affect the upper levels

Data Independence Types

- Logical Data Independence
- Physical Data Independence

Logical Data Independence

- Immunity of external level from changes at conceptual level
- Changes in conceptual model do not affect the external views

Types of Changes

- Adding a new table etc.
- Adding a new field in a table
- Changing the data type/size
- Deleting a field from a table

Physical Data Independence

- Immunity of Conceptual level from changes at Internal level
- Changes in the internal model do not affect the conceptual model

Changes Examples

- Changing file organization
 - Unordered Files
 - Ordered Files
 - Hash Files
- Index implementation
 - Index Sequential Files
 - Secondary Indexes
 - B⁺ tree
- Changing storage medium

Schema

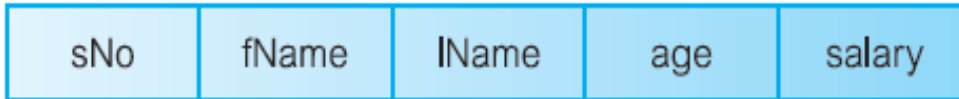
- Greek word (skhēma), which means shape
- The overall description of the database is called the database schema
- Skeleton structure of the database that defines the objects in the database
- A database generally stores its schema in a data dictionary
- In a relational database, the schema defines the schema objects e.g., table, fields, views, indexes, packages, procedures, functions, triggers, sequences, synonyms, materialized views etc.
- Non-schema objects may include: users, roles, contexts etc.
- An Oracle database associates a separate schema with each database user

Mapping

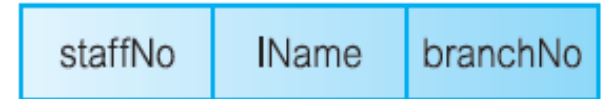
- **External/conceptual mapping.** This enables the DBMS to map names in the user's view on to the relevant part of the conceptual schema
- **Conceptual/internal mapping.** This enables the DBMS to find the physical record (actual record) in physical storage that constitute a **logical record** in the conceptual schema

Differences between the three levels

External view 1



External view 2



Conceptual level



Internal level

```
struct STAFF {
    int staffNo;
    int branchNo;
    char fName [15];
    char lName [15];
    struct date dateOfBirth;
    float salary;
    struct STAFF *next;           /* pointer to next Staff record */
};
index staffNo; index branchNo; /* define indexes for staff */
```

Functions of a DBMS

- **Data Storage, Retrieval, and Update**
- **Concurrency Control Services**
- **Recovery Services**
- **Security and Integrity Services**
- **Support for Data Communication**
- **A User-Accessible Catalog**
- **Utility Services (Backup, Restore, Database tuning)**